

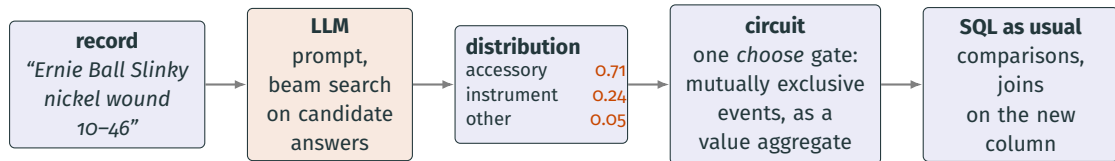
1. Data into neural models

- **TheoremKB**: theorems and proofs out of PDFs; text, fonts, pixels fused, then a sequence model [JCDL 2024]
- **Spreadsheets**: structured cell-type prediction then rule-based table ranges [CIKM 2026]
- **Statistical and data tables**; a table extraction benchmark over 86k pages [KDD 2026]

2. Neural outputs into data systems

- **ProvSQL**, in PostgreSQL: every result carries a **provenance circuit**. Aim: **implement everything** from the literature, over all of SQL [ICDT 2025, ICDE 2026]
- One circuit: provenance, counting, **probabilities**, **Shapley values**; #P-hard but many tractable cases and heuristic tools
- Neural predictions as **uncertain facts**: next slide

Example: keep the model's distribution, not its top answer



10k Amazon musical products, 230k reviews,
augmented locally with Llama 3.1 (8B) [CIKM 2026]:

- contradictions with existing labels show up as **low-probability** answers
- queries **recover** records whose correct label was not the top prediction
- uncertainty propagates **jointly** through joins: a fact used twice is counted once

The point

Reasoning about uncertainty is handled by the system: probabilistic databases, provenance tracking, and circuits.

Topic 1: uncertainty quantification and recalibration of LLM outputs

From log-probs to probabilities

- Token log-probs are **not** the probability of being right. Alternatives: beam exploration, sampling consistency, verbalized confidence. Which one, for which task?
- **Recalibration** against ground truth: per model, per prompt, per domain?
- Beyond categories (**free text**, numbers, spans): what are the mutually exclusive events? How to generate text long enough? How to model the fact that predictions on related inputs are **correlated**?

Into the probabilistic database

- Once grounded, uncertainty is **composed** by the system: joins, aggregates, thresholds, with the right independence assumptions made explicit
- Cost: exact evaluation is #P-hard; which **structure** do LLM-filled tables have, and when is approximation with guarantees enough?
- When the **model changes**, keep the circuit and its evaluation plan, refresh the leaves?

Treat the LLM as one more unreliable source, and store what is needed to stop trusting it later.

Topic 2: representing complex data for neural models

Documents are not text

- **Fonts and layout** carry as much as the words; pixels catch what the text layer loses
- **Long range**: a proof ends where the next theorem begins, pages later. Sequence models over paragraphs beat bigger encoders
- A cheap structured model (e.g., CRF) within a point of a GPU transformer on spreadsheets:
when does structure beat scale? when are we happy with cheap models?

Supervision and evaluation

- **Sources as labels**: use source documents (e.g., $\text{\LaTeX}$ from arXiv) as ground truth; when sources are not available, synthetic sources can build a training dataset
- **Evaluation** is critical: metrics should score the prediction under the predicted uncertainty, not only the top answer

How to exploit neural approaches when they are good, and symbolic approaches when they are cheap and bring semantics?